PART 4.0 INBOUND PROCESSING (SCR-TO-FCR)

4.1 Initial Load Processing Overview

Processing the initial load to the FCR is a one-time activity that will build an outgoing FCR transaction for every case and participant that resides within the SCR. Once the data is extracted from the SCR, an Audit File record should be generated for each transaction. For the purposes of the TAG and the initial load skeleton program, it is assumed that if a participant is attached to more than one case in the SCR, then those participants must be submitted multiple times for each case in which they appear.

The following are the specifications for the initial load and the flow of the skeleton program provided to assist in this function. This is not a required method of processing, but a general guideline if you are not able to use the skeleton code.

***NOTE*:** During the three-week period dedicated to the processing of initial load data, the States have the one-time only option of identifying which Participant records will be proactively matched to the NDNH and FCR. If the individual is matched to the NDNH, States may request New Hire (W4), Quarterly Wage (QW), and/or Unemployment Insurance (UI) data. Refer to the Dear Colleague Letter, DC-98-54 (dated June 16, 1998) for more information. States should enter one of these codes as their desired option. The default will be to return no matches if these positions are not populated. The values for the indicators are:

Position	Value	Description
553	Y	Yes, perform FCR to FCR proactive matching for this person
	N	No, do not perform FCR to FCR proactive matching for this person (default)
554	1	Return W4, QW and UI matches
	2	Return W4 matches only
	3	Return W4 and QW matches
	4	Return W4 and UI matches
	5	Return QW matches only
	6	Return QW and UI matches
	7	Return UI matches only
	8	Return no matches (default)

The initial load process consists of:

Inputs: Files that provide data to the Initial Load Process.

SCR - State Case Registry from which the cases and participants are extracted.

Batchnum File - Holds the batch number that goes in the header record of each file sent to the FCR. This file contains only one record, which consists of a six-digit

number. When creating the transaction file to be sent to the FCR, this file is read to obtain the last batch number issued. The batch number will then be incremented by one (1), moved to the header record, as well as written back to the Batchnum File. This file must be created and an initial Batch Number must be entered. For the initial load, the first two (2) positions of the Batch Number must be "88."

NOTE: If this application resides on an IBM mainframe, this file must be used with DISP=MOD in the DD statement to allow for it to be opened as I/O.

Outputs: Files that are created or updated during the Initial Load Process.

FCR Transaction File – The file sent to the FCR containing all the records for cases and participants that are to be loaded into the FCR in addition to the header and trailer records.

Batchnum File - Same as Input Batchnum File. This will contain the incremented batch number that was submitted to the FCR.

Audit File – The file used to document that the Case and Participant records have been transmitted to the FCR. The file also will register the batch number and date the transactions were sent to the FCR. *Refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for more information on the Audit File structure and layout*.

Processing Flow: Sequence of events that occur during the Initial Load Process.

- Create the Batchnum File.

Initialization:

- Initialize the working storage variables. One variable is the record counter, WS-RECORD-COUNT. This variable is the total number of records on the file; it is required in the trailer record and should be initialized to "2".
- Accept the current date into a working storage variable. This date will be used in the header record and the Audit File.
- Read the Batchnum file, then move the batch number into working storage and the header record. The Batch Number must begin with "88" for the initial load.
- Increment the Batchnum record by one (1) and write to the Batchnum file.
- Write the header record to the transaction file. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for a complete transaction file layout.*

Process SCR Data:

Process Case:

- To process case transactions for the initial load, the SCR will have to be navigated sequentially while extracting the required data elements. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for a complete transaction file layout.*
- Once the required data elements have been read from the SCR, they should be moved to the appropriate positions in the Add Case Transaction. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for a complete transaction file layout.*
- When an Add Case Transaction has been created, the record must be written to the transaction file.
- After the Add Case Transaction record has been written, the record count must be incremented by one (1).
- Build an Audit File record from the case information. *Refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for more information on the Audit File structure and layout*.
- Re-initialize the working storage fields for the Case record.

Process Participant:

- Within each case, the initial extract program should loop through the SCR and extract the required information for each participant. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for the file layouts for the FCR Input Person/Locate Request record layout and the required fields and edits for the FCR Input Person/Locate Request record.*
- Once the required participant data elements have been read from the SCR, they should be moved to the appropriate positions in the Add Participant Transaction. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for the file layouts for the FCR Input Person/Locate Request record layout and the required fields and edits for the FCR Input Person/Locate Request record.*
- When an Add Participant Transaction has been created, the record must be written to the transaction file.
- After the Add Participant Transaction record has been written, the record count must be incremented by one (1).
- Build an Audit File record from the participant information. *Refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for more information on the Audit File structure and layout*.
- Re-initialize the working storage fields for the FCR Input Person/Locate Request record.

Termination:

- When all Case/Participant records have been extracted and written to the transaction file, the trailer record should be built. Initialize the trailer record with the required fields.
- Move the total record count, WS-RECORD-COUNT, to the trailer record. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for the specifications of the trailer record layout and requirements.*
- Write the trailer transaction record to the transaction file.
- Ensure all files used during the process have been closed.

4.1.1 Modifications Required to the Code Skeleton

The code skeleton is provided to the States in an effort to standardize and minimize programming functions that the States must perform. The purpose of the skeleton code is to provide States with all the routine functions required that are specific to the FCR. The skeleton code format requires that the State's programmers tailor the code to their SCR. To accomplish this, the following is a guide to the necessary program modifications.

The COBOL skeleton code for the initial extraction process is provided on the diskette supplied with the TAG. The code should be loaded on the mainframe from the following file:

A:\INITLOAD.TXT

NOTE: All program skeletons provided are stored as text files.

Modifications Required:

Input-Output and File section:

The initial extract program is set up to have an input record that is in the same record format as the output transaction record layout. This is done primarily as a placeholder and for testing purposes. The input record needs to be taken out when the code to navigate the State's CSE system is inserted. The code that needs to be removed is as follows:

In the FILE-CONTROL take out the line that reads:

SELECT SCR-INPUT-FILE ASSIGN TO UT-S-SCRINPUT.

In the FILE SECTION take out the following code:

```
FD SCR-INPUT-FILE
RECORDING MODE IS F
LABEL RECORDS ARE STANDARD
BLOCK CONTAINS 0 RECORDS
DATA RECORD IS SCR-INPUT-RECORD.
01 SCR-INPUT-RECORD        PIC X(640).
```

The AUDIT-FILE entry should be set up as a keyed file, most likely a VSAM file; however, for the purposes of this program, there will be no reads issued against this file. *Please refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for the specifics of the Audit File and the key structures required for this file.*

Working Storage:

The skeleton code will move certain values from working storage fields into the header record. These variables will need to be set to the State-specific values.

For the initial load program, the only working storage variable that needs to be set is the State FIPS code. This can be found in the following location:

```
01 WS-LITERALS-AND-COUNTERS.
05 WS-STATE-FIPS     PIC X(2) VALUE '**'.
```

The working storage section includes a table that shows the two character FIPS codes for all States and territories. The '**' value needs to be replaced with the appropriate code for the State.

Another modification to working storage is to insert the record formats for the CSE system. The only records that should be required are the Case and Participant records. (This is a general rule and may not be applicable in every State.) The other record that may require special consideration is the Family Violence Indicator.

Procedure Division:

The primary functions of the procedure division are already coded. These functions include: creating the header record, performing the general edits, formatting the record, creating the transaction records, creating the trailer record, and creating of the audit transaction record.

The required code changes for this program are primarily the navigation of the CSE system and mapping of the data elements to the working storage variables. The

program performs all edits and formats the output transaction record based on the values in these working storage fields. The modifications required are as follows:

NOTE: All alpha and alpha/numeric fields should be left justified with no leading spaces.

In paragraph 1000-PROCESS-CASE:

Write the read statement to navigate to the Case record of the CSE system. After the Case record has been read, paragraph 1100-MAP-CASE-RECORD is performed to move the values to the working storage variables and perform edit checks. Paragraph 1100-MAP-CASE-RECORD will need to be modified to map the State's case fields to these elements.

In paragraph 1050-PROCESS-PARTICIPANT:

The read statement needs to be written to navigate to the Participant record of the CSE system. It is assumed that navigation to the Participant record has to go through the Case record; this may or may not be the case in each State. This portion of the extract program may need to be modified to reflect the correct navigation for a unique State situation. Once the Participant record has been read, paragraph 1150-MAP-PARTICIPANT-RECORD is performed to move the values to the working storage variables against which the edits will be performed. Paragraph 1150-MAP-PARTICIPANT-RECORD will need to be modified to map the State's participant fields to these elements.

4.1.2 JCL Requirements

The following is an example of the JCL required to execute the initial extract skeleton program. The TAG targets SCR systems that reside on an IBM mainframe. For systems that are on different platforms, the TAG should serve as a model for the State's required development. States on other platforms may receive technical assistance for their specific systems through the Office of Child Support Enforcement (OCSE). *Please refer to Appendix A, "Assistance Contacts," for names, phone numbers and e-mail addresses.*

```
//JOBNAME JOB '<ACCOUNTING INFO>',
//    MSGCLASS=A
//STEP1   EXEC PGM=INITEXTR
//SYSPRINT DD SYSOUT=*
//SYSOUT  DD SYSOUT=*
//PRINTER  DD SYSOUT=*
//FCROUT  DD DSN=<FCR TRANSACTION FILE DSN>,DISP=(,CATLG,DELETE),
//       UNIT=SYSDA,SPACE=(CYL,(100,100),RLSE),
//       DCB=(RECFM=FB,LRECL=640,BLKSIZE=32000)
//FCRBATCH DD DSN=<BATCHNUM FILE DSN>,DISP=MOD
```

```
//AUDITFILE DD DSN=<AUDIT FILE DSN>,DISP=MOD
```

Listed below is an example of the JCL required to create the Batchnum File:

```
//JOBNAME JOB '<ACCOUNTING INFO>',MSGCLASS=A
//STEP1    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSIN  DD DUMMY
//SYSUT1  DD *
880000
//SYSUT2  DD DSN=<BATCHNUM FILE DSN>,DISP=(,CATLG,DELETE),
//      UNIT=SYSDA,SPACE=(TRK,(1,1),RLSE),
//      DCB=(RECFM=FB,LRECL=6,BLKSIZE=6)
```

NOTE: For the initial load of SCR data, the Batch Number must begin with "88".

4.2 Processing Routine Transactions Overview

Processing the routine load to the FCR is an ongoing activity that builds an outgoing FCR transaction for each case and participant in the Trigger File. *Please refer to Section 2.1, "Explanation of the Trigger File Concept," of this TAG for an explanation of the Trigger File.* When the Trigger File is processed, FCR transactions will be generated and the Audit File records created.

For the purposes of this document and the routine processing skeleton program, it is assumed that navigation through the State CSE system is via the Case record, then to its corresponding Participant records. It is also assumed that non IV-D orders are stored with the IV-D cases. If this is not the case, another input file will need to be identified for the non IV-D orders.

Following are the specifications for the routine transactions and the flow of the provided skeleton program. This is not a required method of processing, but a general guideline if the State is not able to use the skeleton code.

For record layouts of the transaction records, please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts."

The Routine Process consists of:

Inputs: Files that provide data to the Routine Process.

> **Trigger File** - The Trigger File contains the key(s) to Case/Participant records that should be extracted from the SCR and forwarded to the FCR. The Trigger File contains the action taken by the worker and the key(s) necessary to navigate the SCR

when extracting the required data. *Please refer to Section 2.1, "Explanation of the Trigger File Concept," of this TAG for an explanation of the Trigger File.*

SCR - This is the State Case Registry from which the cases and participants are extracted.

Batchnum File - The file that contains the batch number used in the header record. This file contains only one record, which consists of a six-digit number. When creating the transaction file to the FCR, this file is read to obtain the last batch number issued. The batch number will then be incremented by one (1), moved to the header record, and written back to the Batchnum File.

NOTE: If this application resides on an IBM, the file must be used with DISP=MOD, in the DD statement to allow for it to be opened as I/O.

Outputs: Files that are created or updated during the Routine Process.

FCR Transaction File – The file that is sent to the FCR containing all of the records for cases and participants to be loaded into the FCR, along with a header and trailer record.

Batchnum File - Same as Input Batchnum File. This will contain the incremented batch number that was submitted to the FCR.

Audit File – The file used to document that the Case and Participant records have been transmitted to the FCR. The file will also register the batch number and date the transactions were sent to the FCR. *Refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for more information on the Audit File structure and layout*.

Processing Flow: The sequence of events that occur during the Routine Process.

Initialization:

- Initialize the working storage variables. One variable is the record counter, WS-RECORD-COUNT. It is the total number of records on the file and is required in the trailer record. It should be initialized to "2."
- Accept the current date into a working storage variable. This date will be used in the header record and the Audit File.
- Read the Batchnum file, then move the batch number into working storage and the header record.
- Increment the Batchnum record by one (1) and write to the Batchnum file.
- Write the header record to the transaction file. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for a complete transaction file layout.*

Process Data:

Process Trigger File - This file is the repository of all actions taken by workers against the SCR. The Trigger File will be read to obtain the type of action taken and the key(s) to the Case or Participant record. Once a record is read from the Trigger File, a key will be created to navigate the SCR, necessary data will be extracted to build an FCR transaction, and the Audit File will be updated. *Please refer to Section 2.1, "Explanation of the Trigger File Concept," of this TAG for a complete list of transaction types.*

Process Case:

- Extraction of case information will occur when a record from the Trigger File has been matched to the SCR. The program will navigate the SCR to extract the necessary data elements by moving them to the working storage fields. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for a complete transaction file layout. Also refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for more information on the Audit File structure and layout.*
- The working storage fields that have been populated in the previous step will be moved to the FCR transaction output record and the Audit File transaction output record.
- After the FCR transaction output record has been created, write the record to the transaction file and increment the total record count. If the skeleton code is being used, this variable is WS-RECORD-COUNT and will be incremented when the record is written.
- After the Audit File transaction output record has been created, write the record to the Audit File. ***NOTE:*** There is no record counter associated with this file.
- Re-initialize the working storage variables for the Case record.

Process Participant:

- To extract participant data, the routine transaction program will use the keys from the trigger record to navigate through the SCR Case record to its Participant records. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for the FCR Input Person/Locate Request record layout and the required fields and edits for the FCR Input Person/Locate Request record.*
- When the Participant record is found in the SCR, move the required data elements to the working storage variables. As the Participant record is being accessed, the program should check the status of the participant. If a locate is required for the participant, the *Locate-Source* data elements should be populated. ***NOTE:*** This will initiate an *FPLS locate* on the individual.
- Move the data from the working storage variable to the FCR transaction output record and audit transaction output record.

- Once the Participant record has been created, write the record to the transaction file and increment the total record count. If the skeleton code is being used, this variable is WS-RECORD-COUNT and will be incremented when the record is written.
- After the Audit File transaction output record has been created, write the record to the Audit File. ***NOTE:*** There is no record counter associated with this file.
- Re-initialize the working storage variables for the Participant record.

Termination:

- When all Case/Participant records have been extracted and written to the transaction file, the trailer record should be built. Initialize the trailer record with the required fields.
- Move the total record count, WS-RECORD-COUNT, to the trailer record. *Please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," for the specifications of the trailer record layout and requirements.*
- Write the trailer transaction record to the transaction file.
- Ensure all files used during the process have been closed.

4.2.1 Modifications Required to the Code Skeleton

The code skeleton is provided to the States in an effort to standardize and minimize the amount programming functions that the States must perform. The purpose of the skeleton code is to provide States with all the routine functions required that are specific to the FCR. The skeleton code format requires that the State's programmers tailor the code to their SCR. The following is a guide to the necessary program modifications.

The COBOL skeleton code for the routine transaction extraction process is provided on the diskette supplied with the TAG. The code should be loaded onto your mainframe from the following file:

A:\STRIPFCR.TXT

NOTE: All program skeletons provided are stored as text files.

Modifications Required:

Input-Output section and File section:

The routine transactions extract program is set up to have an input record in the same record format as the output transaction record layout. This is done primarily as a placeholder and for testing purposes. This file needs to be removed when the code to

navigate the State's CSE system is inserted. The code that needs to be removed is as follows:

In the FILE-CONTROL take out the line that reads:

SELECT SCR-INPUT-FILE ASSIGN TO UT-S-SCRINPUT.

In the FILE SECTION take out the following code:

```
FD SCR-INPUT-FILE
RECORDING MODE IS F
LABEL RECORDS ARE STANDARD
BLOCK CONTAINS 0 RECORDS
DATA RECORD IS SCR-INPUT-RECORD.
01 SCR-INPUT-RECORD        PIC X(640).
```

In these sections, the AUDIT-FILE entry should be set up as a keyed file, most likely a VSAM file; however, for the purposes of this program, there will be no reads issued against this file. *Please refer to Section 2.2, "Explanation of the Audit File Concept," of this TAG for the specifics of the Audit File and the key structures required for this file.*

Working Storage:

The skeleton code will move certain values from working storage fields into the header record. These variables will need to be set to the State specific values.

For the routine transactions load program, the only working storage variable that needs to be set is the State FIPS code. This can be found in the following location:

```
01 WS-LITERALS-AND-COUNTERS.
05 WS-STATE-FIPS     PIC X(2) VALUE '**'.
```

There is a table in the working storage section that shows the two character FIPS codes for all States and territories. The '**' value needs to be replaced with the appropriate code for the State.

The record formats for the CSE system need to be inserted into the working storage section as well. The only records that should be required are the Case record and the Participant record. (This is a general rule and may not be applicable in every State.)

Other records that may require special consideration are the Family Violence Indicator and the Locate Status for the participant.

Procedure Division:

Most of the procedure division's primary functions are already coded. These functions include: creating the header record, performing the general edits, formatting the record, creating the transaction records, creating the trailer record, and creating the audit transaction record.

The required code changes for this program are primarily the navigation of the CSE system and mapping of the data elements to the working storage variables. The program performs all edits and formats the output transaction record based on the values in these working storage variables. The modifications required are as follows:

NOTE: All alpha and alpha/numeric fields should be left justified with no leading spaces.

In paragraph 1000-PROCESS-CASE:

The read statement needs to be written to navigate to the Case record of the CSE system. This is accomplished by utilizing the Case Number supplied in the trigger record. After the Case record has been read, paragraph 1100-MAP-CASE-RECORD is performed to move the values to the working storage variables and perform edit checks. Paragraph 1100-MAP-CASE-RECORD will need to be modified to map the State's case fields to these elements.

In paragraph 1050-PROCESS-PARTICIPANT:

The read statement needs to be written to navigate to the Participant record of the CSE system. It is assumed that navigation to the Participant record has to go through the Case record; this may or may not be the case in each State. This portion of the extract program may need to be modified to reflect the correct navigation for a unique State situation. The Case Number and the member identification are supplied in the trigger record to achieve this navigation. Once the Participant record has been read, paragraph 1150-MAP-PARTICIPANT-RECORD is performed to move the values to the working storage variables against which the edits will be performed. Paragraph 1150-MAP-PARTICIPANT-RECORD will need to be modified to map the State's participant fields to these elements.

Prior to performing this paragraph, the working storage variable that indicates that a LOCATE is being requested will need to be set. To accomplish this, the following statement must be modified to reflect the locate flag set by the State CSE system.

IF <*STATE SPECIFIC LOCATE INDICATOR*> = 'LOC'
SET WS-ISSUE-LOCATE TO TRUE

END-IF.

The default request for locate sources will be ALL sources; if a State does not wish to request all sources, a modification must be made in paragraph 1350-WRITE-PARTICIPANT-RECORD. *For a complete list of locate sources, please refer to the FCR IGD, Appendix G, "FCR Input Transaction Layouts," under FCR Input Person/Locate Request Record for Locate sources.*

4.2.2 JCL Requirements

The following is an example of the JCL required to execute the routine extract skeleton program. This manual targets systems that reside on an IBM mainframe. For systems on different platforms, the TAG should serve as a model for the required development. States on other platforms may receive technical assistance for their specific systems through OCSE. *Please refer to Appendix A, "Assistance Contacts," of this TAG for contact names, phone numbers and e-mail addresses.*

```
//JOBNAME JOB '<ACCOUNTING INFO>',
//    MSGCLASS=A
//STEP1   EXEC PGM=SCREXTR
//SYSPRINT DD SYSOUT=*
//SYSOUT  DD SYSOUT=*
//PRINTER  DD SYSOUT=*
//TRIGGER  DD DSN=<TRIGGER FILE DSN>,DISP=SHR
//FCROUT  DD DSN=<FCR TRANSACTION FILE DSN>,DISP=(,CATLG,DELETE),
//       UNIT=SYSDA,SPACE=(CYL,(100,100),RLSE),
//       DCB=(RECFM=FB,LRECL=640,BLKSIZE=32000)
//FCRBATCH DD DSN=<BATCHNUM FILE DSN>,DISP=MOD
//AUDITFILE DD DSN=<AUDIT FILE DSN>,DISP=MOD
```

Listed below is an example of the JCL required to create the Batchnum file:

```
//JOBNAME JOB '<ACCOUNTING INFO> ',MSGCLASS=A
//STEP1    EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSIN  DD DUMMY
//SYSUT1  DD *
000000
//SYSUT2  DD DSN=<BATCHNUM FILE DSN>,DISP=(,CATLG,DELETE),
//      UNIT=SYSDA,SPACE=(TRK,(1,1),RLSE),
//      DCB=(RECFM=FB,LRECL=6,BLKSIZE=6)
```

NOTE: This file will be created for the routine processing and the batch number will **not** begin with "88."